Principles of programming - opis przedmiotu

Informacje ogólneNazwa przedmiotuPrinciples of programmingKod przedmiotu11.3-WE-AutP-PrinProgr-ErWydziałWydział Nauk Inżynieryjno-TechnicznychKierunekAutomatyka i robotykaProfilogólnoakademickiRodzaj studiówProgram Erasmus pierwszego stopniaSemestr rozpoczęciasemestr zimowy 2021/2022

Informacje o przedmiocie

······································			
Semestr	1		
Liczba punktów ECTS do zdobycia	5		
Typ przedmiotu	obowiązkowy		
Język nauczania	angielski		
Sylabus opracował	• dr inż. Grzegorz Łabiak		

Formy zajęć

) =						
Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia	
Wykład	30	2	-	-	Egzamin	
Laboratorium	30	2	-	-	Zaliczenie na	
					ocenę	

Cel przedmiotu

The goal of the subject is to teach programmining in C language. In the first place the learning consists in teaching syntax and semantics of C language. Next, students are instructed how to implement simple computational algorithms. Apart from that some elements of computational complexity are introduced, which allows to asses computational cost of implemented solutions.

Wymagania wstępne

Fundamentals of mathematics

Zakres tematyczny

Introductory information: a program and its components
Pogramming environment. Source files. Compilation. Basic elements of program and its structure. Main function. Functions and procedures.
Basic types. Variables. In/out operations. Operators.
Iterative loops: for, while, do-while
Decision instructions.
Creation of own Functions.
Arrays and character strings.
Structures.
Pointers.
Sorting algorithms: buble sort, selection sort, quick sort.

Metody kształcenia

Lecture, laboratory exercises

Efekty uczenia się i metody weryfikacji osiągania efektów uczenia się

Opis efektu	Symbole efektów Metody weryfikacji	Forma zajęć	
Student is able to solve algorithmic problem, implement it in C language, run and test	 bieżąca kontrola na 	 Laboratorium 	
it.	zajęciach		
	• sprawdzian		
Student is able to define real world problemes and present them as computational	 bieżąca kontrola na 	 Laboratorium 	
problmes, critialy analise solutions and assess computational complexity	zajęciach		
	• sprawdzian		
Student is able to implement typical algorithms, such as sorting, searching, etc.	• kolokwium	• Wykład	
		 Laboratorium 	

Opis efektu	Symbole efektów Metody weryfikacji	Forma zajęć
Student knows the difference between structural and object oriented programming	 bieżąca kontrola na zajęciach 	• Laboratorium

Warunki zaliczenia

Lecture - exam - in order to get a credit it is necessary to pass all of the required tests (oral or written) Laboratory - the main condition to get a pass are sufficient marks for all exercises and tests conducted during the semester Calculation of the final Grade: lecture 50% + laboratory 50%

Literatura podstawowa

- 1. Kernighan B. W., Ritchie D. M.: Język Ansi C, WNT, Warszawa, 1994.
- 2. Sielicki A.: Laboratorium programowania w języku Pascal, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 1994

Literatura uzupełniająca

- 1. Aho A. V., Hopcroft J. E., Ullman J. D.: Projektowanie i analiza algorytmów, Helion, Warszawa, 2003.
- 2. Banachowski L., Diks K., Rytter W.: Algorytmy i struktury danych, WNT Warszawa, 2001.
- 3. Roszkowski J.: Analiza i projektowanie strukturalne, Helion, Gliwice, 2002.
- 4. Wirth N.: Algorytmy + struktury danych = programy, WNT, Warszawa, 1989.

Uwagi

Zmodyfikowane przez dr hab. inż. Wojciech Paszke, prof. UZ (ostatnia modyfikacja: 12-07-2021 07:56)

Wygenerowano automatycznie z systemu SylabUZ